

Actionable!

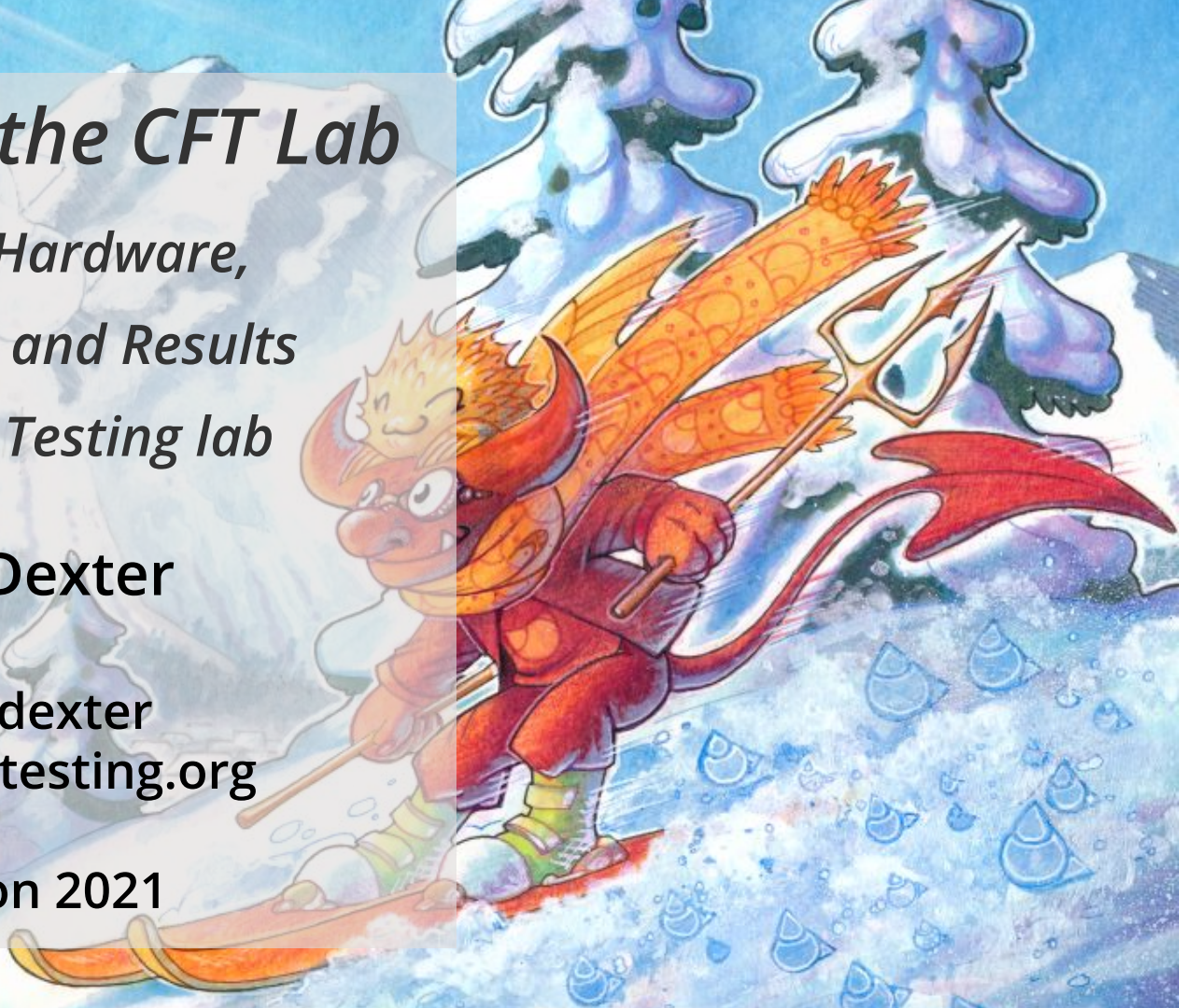
Lessons from the CFT Lab

*A tour of the Hardware,
Methodologies, and Results
of the Call For Testing lab*

Michael Dexter

@michaeldexter
editor@callfortesting.org

EuroBSDcon 2021



Menu – *Are you in the right room?*

- Thank you EuroBSDcon!
- Motivations
- A Brief *blah blah blah* about *blah blah blah*
- Actionable Hardware Lessons
- Actionable Software Lessons
- The Build Option Rabbit Hole – With results!
- Thank you EuroBSDcon!

Thank you EuroBSDcon!

- 2001: Moved to Latvia, burned by RPM Hell
- 2002: Discovered FreeBSD 4.7 Jail(8)
- 2003: `cvsup.free|net|open.bsd.lv` mirrors
- 2006: EuroBSDcon Milan
- 2007: EuroBSDcon Copenhagen
- 2008: Strasbourg – First conference talk

Paldies Pēter!

Paldies Kristap!

Thank you EuroBSDcon!

- 2011: Maarssen – Talk, announced Call For Testing
- 2012: Warsaw – Talk
- 2015: Stockholm – Talk
- Plus OpenCON, FOSDEM, LinuxTAG, systems.de...
- Hosted PikeCon, Peter Hansteen, MySQL in Rīga...
- 2021: Vienna – Talk...

Thank you EuroBSDcon!

- 2009 – Present – Portland Linux/Unix Group
- *<Insert a disturbing number of talks and efforts>*

*You and Ryanair .99 € plus tax
flights are to ~~blame~~ thank!*

Plus countless wonderful people.



Pro Tip: Embrace the Formality – AsiaBSDCon!

- Imagining or finishing projects and communicating them to other humans is a valuable skill
- Formal paper writing skills are good for the brain and hopefully your day job
- I guarantee I was more shy than you. Yes, really.
- Again, Thank You!

Inspiration and Motivation

*Reflections During the
Crappy Sabbatical*

Inspiration... Visceral inspiration

- January 1991: BSD Unix in college
 - Unix: Inside the Computer/Behind the Camera
- 2008 RedHat 5.2 RPM Hell
- This “Open Source” Thing
 - Proprietary software can rarely be fixed
 - Feature suggestions may be *discouraged*
 - Frequently discontinued or cloudified

Motivations: Why a Lab?

- Open Source is participatory
 - Proprietary platforms can meet many needs
 - Hardware provides fundamental isolation
- Practical isolation...

Honey! What do you think of HaikuOS?

I HAVE A CONFERENCE CALL IN TEN MINUTES!

Motivations Revisited – From my 2008 slides

Zen and the Art of **Multiplicity** Maintenance

Motivations

RPM Hell and user privacy

Solution: separation, compartmentalization, containment, imprisonment or *isolation* of filesystems, applications and/or users.

You kids get off my file system!

Motivations – From my 2008 slides Cont.

Zen and the Art of **Multiplicity** Maintenance

Motivations

Cross-platform development

Please, one keyboard. Even if the package
needs to be build under six OS's.

The *consolidation* of systems

Motivations – From my 2008 slides Cont.

Zen and the Art of **Multiplicity** Maintenance

Motivations

Moving the @

1991 – Hundreds of simultaneous users on a
33MHz Sun 4/490 (first dexter@)

2009? – Hundreds, if not thousands of
isolated hosts on a 3GHz i386

Motivations – From my 2008 slides cont.

Zen and the Art of **Multiplicity** Maintenance

Confession

I just want my damn computers
to work reliably

Motivations → Focus → Talks

Revised after all these years:

I consider the ***authorized and validated synchronous write within hardware and virtual machines*** to be the highest priority of the industry and community

In short: Virtual machines + OpenZFS + NVMe

A Brief *blah blah blah*
about *blah blah blah*

A Brief *blah blah blah* about *blah blah blah*

- The 20/80 rule of Open Source user satisfaction
- Not to be confused with the 80/20 principle
 - 20% of inputs give rise to 80% of outputs...
- Heard it from Allan Jude, not sure who coined it
- Basically: “OMG! The hypervisor booted!”



A Brief *blah blah blah* about *blah blah blah* Cont.

- Users cheer *and* bemoan your exciting new project at 20% completion because it's not yet useful
- Users bemoan your exciting new project at 80% because it's... *only* 80% complete



Let's talk about that...

- If you fear a mid-life crisis, you'll *hate* a full-life crisis
- 100% Software: UnixWare, FreeHand, FileMaker 4.1
- Twilight Software: OpenOffice.org! Pike? Python 2!

MeetBSD 2010: "We need a hypervisor"

`diskctl(8)` – PURE VAPORWARE!

"Some kinda dump for CCBs!" – VAPORWARE!!!

Let's talk about that...

The first 1% of software development
is just as important as the last 1%,
along with every step in between

Let's talk about that...

*Why, sometimes I've believed as many as six
impossible things before breakfast.*

— *Lewis Carroll, Through the Looking Glass*

Mea culpa!

Select, Actionable Hardware Lessons

*Buckle Up
Buttercup*

Actionable Hardware Lessons

- Home Labs discussion at this conference
- Log and info at callfortesting.org
- Hopefully this information is... deduplicated and

You learn something new!

Hardware Lesson 1: Dream Big

- Maintain your dream lab on paper
- You will learn from every bit of research you do
- Learn to recognize deals when you see them
 - Move quickly when you see them
 - Budget permitting...
- Used hardware pricing often defies logic

Hardware Lesson 2: Embrace Placeholders

- That free first server and switch are the best in the world for a brief moment
- You *will* learn from them, even if the lesson is to never to own one of either again

Hardware Lesson 3: Ally #1: Recycling Centers

- Find a local center that accepts and sells e-waste
- They might keep a watch list for you
- They may sell on eBay but allow for local pickup
- They'll take it back for free when you're done!

Hardware Lesson 4: Ally #2: Electricians for UPSs

- *Buy UPS batteries, not Uninterruptible Power Supplies*
- Not all UPSs are “computer components”
- Many are replaced early on a regular basis
- Befriend a local Electrical Engineering firm
- Warning: You may have to take all of them
 - See: Ally #1: e-waste recycling centers

Hardware Lesson 4: Ally #2: Electricians Cont.

- The power conditioning alone is worth it
- 5 minutes of backup is better than zero
- Puts battery replacement cost on your terms
- Learning different UPSs is just as important as learning other components!

Hardware Lesson 5: Stay Consistent/Buy Two

- Some lab work requires diversity but most lab work requires consistency/reproducibility
- Have a lab sibling of your production system
- “Lot” pricing can also defy logic
 - One genuine Amphenol SATA cable: 16 €
 - 25 genuine Amphenol SATA cables: 16 €

Convenience Store Pricing?

Hardware Lesson 5: Stay Consistent Continued

- 1998ish: 2 X Pentium II's
- 2005ish: 2 X Via 1GHz Mini-ITX
- 2008ish: 2 X Intel D845GBV + Pentium IV 2.4GHz
- 2010ish: Rando Free Servers
- 2010ish: Rando pre-SandyBridge ThinkPads

Hardware Lesson 5: Stay Consistent Continued

- 2020ish: N X HP Z220, Z420, Z820
 - Z220 Xeon E3 costs less than an i5 ㄟ_(\ツ)_/
- 2020ish: 2 X Compellent/R720
- New HP EPYC DL 325
- 2021+ Recertified Dell R730xd's



Hardware Lesson 6: Noise and Power Draw

- HUGE progress has been made in these areas
- Nearly 10 years of quiet systems
- “Green” movement
- Note fans on switches
- Few second-hand fanless computers though

Hardware Lesson 7: Invest in consistent cables!

- Random computers = random cables
- Invest in consistent network and power cables!
- Make this your first 50 € to 100 € purchase
- Quality cables will outlast generations of CPUs
- You will thank yourself for many years



Hardware Lesson 8: Invest in consistent screws!

- Can't afford 50 € in cables yet?
- Invest in case, drive, and ThinkPad screws!
- The cool black ones do not cost extra...
- Note that HPE caddy screws are a bit longer...
- You will thank yourself!



Hardware Lesson 9: Power Distribution Units...

- A “shop” power strip and zip ties are very useful
- Budget PDUs exist but TEST EVERY PORT (don’t ask)
- Great compliment to those consistent power cords



Hardware Lesson 10: “Racks”

- A full-height cabinet is rarely useful
- Rolling carts with two to four shelves are great
- Short “telco” racks are very useful
- The Ikea LACK Rack makes a great *story*...



Hardware Lesson 11: Storage Devices

- 2020: Invest in 10 to 20 identical 120GB SSDs...
- 2021: Good luck out there!
- All HDDs are terrible in some way
- Navigate the shortest path to U.2 NVMe storage
- SATA and SAS SSDs should have never happened...

Hardware Lesson 12: The Other Storage Devices

- Another great 25 € to 100 € investment
Is in IKEA storage tubs
- Many sizes
- Transparent
- Don't forget the lids!



Hardware Lesson 13: Network Infrastructure

- Yes.
- Rando managed switches are often free/affordable
- Invest in a “dumb” 48 port switch or two
 - Always useful
 - Consistent, Simple, Reliable, no Patch Tuesday

Hardware Lesson 14: Document as you go

- You will thank yourself for documenting...
 - System Serial numbers, License key
 - IP Addresses
 - Firmware versions, default firmware passwords
 - RAM module part numbers
 - Drive models, serial numbers, firmware

Hardware Lesson 14: Document as you go Cont.

- You will thank yourself for documenting...
 - Data locations
 - Installer image/ISO locations
 - Invaluable when it's time to *grow* your lab
 - Make a habit of it
 - Make it easy – Consider collaborative docs

Hardware Lesson 15: Test consistency/reliability

- Test those cables, even the low-tech ones, sadly

“The person who made this cable has never used a computer”

- Test power draw
- Test noise
- Test for equal performance from “identical” systems
- Ideally log everything cradle to grave

Hardware Lesson 16: Anything can fail

- Is it hardware? It can fail.
- Never rule out anything failing. Ever.
- “New” and “expensive” does not preclude failure
 - Often quite the opposite – DOA
- Be nimble switching between systems and components when diagnosing issues

Hardware Lesson 17: Ergonomics matter

- Personal Protective Equipment is not expensive
- Medical treatments can be very expensive
- Disposable gloves are handy
- Every keyboard is terrible in some way
- Invest in your sitting/standing hardware

Hardware Lesson 18: Protect your hearing

- I *love* my 3M Peltor hearing protection
- I *live* on my wired AfterShokz Sportz Titanium Open Ear Wired Bone Conduction Headphones



Thank you @BlueCollarMage!



Hardware Lesson 19: Learn, learn, learn

- *Every piece of hardware will teach you something*
- *Colleagues are invaluable resources*
- *Knowing what **not** to buy is just as important*
- *Note the Home Labs session, Reddit...*
- *Pay it forward! Give away what you outgrow*
- Full Lab Notes at callfortesting.org/lab

Hardware Lesson 20: My Secret Weapon!

Painter's Tape and a Sharpie



Select, Actionable Software Lessons

Whew!

Software Lesson 1: Own the stack

- The BSDs, followed closely by illumos deliver...
 - Software Freedom/Open Source
 - Invaluable consistency and unity
 - Unix, CTRL-T, OpenZFS, DTrace, bhyve, jail | zones
 - Upstreams of OpenSSH, PF, mandoc, openrsync...

An unrivaled, highly-flexible, and repairable toolkit

Software Lesson 2: “Everyone knows that”

- CTRL-T: Show status/progress of a command line utility
- FreeBSD: New package “missing”? – `rehash`
- *30+ ThinkPad scroll lock: `Fn-CTRL-K` (up/down arrow)
- Panic the system: `sysctl debug.kdb.panic=1`

Why didn't you tell me you could do that!?!

Software Lesson 2: “Everyone knows that” Cont.

- Exit single user mode on boot: `exit`
- Enter single user mode booted: `shutdown now*`
- Exit single user mode on boot: Also `exit`
- Enter debugger (if enabled): `CTRL-ALT-ESC`
- `etcupdate extract` after updating for a 3-way merge
- `ls /usr/share/man/man8/ ... mandoc(1) or zgrep(1)`

Thank you Klara Systems!

Software Lesson 3: OpenZFS tips

- `zfs get written` – Get one property for all datasets!
- `zpool import -N` – Import without mount for replication!
- `zfs diff <snapshot> <snapshot>` – Delta log!
- `zpool import -n` – Dry run! Could be the default!
- `zfs set readonly=on ...` – Effortless immutability!
- `zfs set snapdir=visible` – History transparency!

Software Lesson 4: Laptop Battery Health

```
# acpiconf -i 0
```

```
Design capacity:      23940 mWh
```

```
Last full capacity:   20270 mWh
```

```
...
```

```
# acpiconf -i 1
```

```
Design capacity:      24050 mWh
```

```
Last full capacity:   20490 mWh
```

```
...
```

Software Lesson 5: Memory Errors: dmidecode

```
MCA: Bank 10, Status 0x8c000050000800c2
```

```
MCA: Address 0x13b511e680
```

```
Handle 0x002D, DMI type 20, 35 bytes
```

```
Memory Device Mapped Address
```

```
    Starting Address: 0x000000000000
```

```
    Ending Address: 0x001FFFFFFFFF
```

```
    Range Size: 8 GB
```

The address is simply an offset within a range

Software Lesson 6: FreeBSD/ZFS/Samba 4 AD

Yes, FreeBSD 13, OpenZFS, and Samba 4 can
provide a basic Active Directory server

`github.com/michaeldexter/freebsd-ad`

Software Lesson 7: FreeBSD Xen Host

Yes, FreeBSD 13 is a promising Xen Host

`github.com/michaeldexter/xenomorph`

Software Lesson 8: smartctl(8) output and jq

```
smartctl -ij /dev/ada0 | jq -rj \  
  '[' " ", [.model_name], [" "], [.serial_number], [" "],  
  [.firmware_version], [" "], [.form_factor | .name], [" "],  
  [.rotation_rate], ["RPM "], [.logical_block_size], [" "],  
    [.physical_block_size], [" "]  
  | .[]'
```

```
WDC WD80EFAX-68KNBN0 TMUX05ZK 81.00A81 3.5 inches 5400RPM 512 4096
```


Software Lesson 9: DTrace for the rest of us

```
dtrace -q -n 'proc:::exec-success { printf("%d %d\n", uid, walltimestamp , curpsinfo->pr_psargs); }'
```

```
0 1631668500003756095 /bin/sh -c /usr/libexec/atrun
```

```
0 1631668500005754862 /usr/libexec/atrun
```

Note that piped commands show up as two commands

Software Lesson 10: Extract that Windows Key

FreeBSD: `pkg install acpica-tools`

`acpica-tools-20210604` Tools from the ACPI Component Architecture
(ACPICA) project

```
/usr/local/bin/acpidump | \  
grep -A5 MSDM | \  
tail -n3 | cut -c60-75 | \  
xargs echo | sed -e 's/^\.*//' -e 's/ //g'  
HGCNP-BHYVE-7MBB9-CHMOD-C9XTP
```

Down the Build Option Rabbit Hole 2003 ~ 2021

Own. The. Stack.



Software Lesson 11: Reflect for a moment...

FreeBSD 13.0 was a milestone release

- Upstream OpenZFS
- Reproducible Builds
- Pulled from Git
- Working src.conf(5) Build Options
- I REPEAT, WORKING BUILD OPTIONS!

Software Lesson 11: Reflect for a moment Cont.

2003 Epiphany: Minimum Jails, come VMs

- Satisfy software dependencies and little more (l_{dd}(1) !)
- You can't attack what is not installed
- Truly do one, and only one thing well
- Want a fast build? BUILD LESS!
- Almost as if it is a form of *software containment...*

Software Lesson 12: Build Options & KERNCONFs

```
man src.conf
```

```
WITHOUT_ACCT
```

```
Set to not build process accounting tools such as accton(8) and  
sa(8).
```

```
WITHOUT_ACPI
```

```
Set to not build acpiconf(8), acpidump(8) and related programs.
```

```
WITH_...
```

230+ Options in total for 13.0

Software Lesson 13: Build Options & KERNCONFs

```
make -C /usr/src showconfig  
  
MK_VI          = yes  
  
/usr/src/rescue/rescue/Makefile  
  
.if ${MK_VI} != "no"  
  
CRUNCH_PROGS_usr.bin+= vi  
  
CRUNCH_ALIAS_vi= ex  
  
.endif
```

Software Lesson 14: Like a BOS... bos-ng

Thank you PHK!

/usr/src/tools/tools/build_option_survey

WITHOUT_CXGBETOOL=foo	MK_CXGBETOOL = no	+0	-0	*1577	3426728	-100	+0	-3	*1	3423676	-3152	+0	-3	*1578	3423576	-3252
WITHOUT_CXX=foo	MK_CLANG = no MK_CLANG_FULL = no MK_CXX = no MK_GOOGLETEST = no MK_LLVM_COV = no MK_TESTS = no	failed					+0	-8501	*3	1889620	-1537208	+0	-8501	*1186	1889580	-1537248
WITHOUT_DEBUG_FILES=foo	MK_DEBUG_FILES = no	failed					+0	-1902	*21	2054824	-1372004	+0	-1932	*2923	1519148	-1907680
WITHOUT_DIALOG=foo	MK_BSDINSTALL = no MK_DIALOG = no	failed					+0	-222	*1	3416836	-9992	+0	-222	*1566	3416704	-10124

Software Lesson 15: Like a BOS... bos-ng

`github.com/michaeldexter/bos-ng`

`findfaults.sh`

`grep -m 1 -B 10 stopped buildfail.txt`

Software Lesson 16: Enter OccamBSD...

Nudge. Grumble. Report...

OMG! Thank you Kyle Evans!

Mitchell Horne!

Björn Zeeb!

Ed Maste!

They're working! (Within limits and well enough)

Software Lesson 17: Reduce to `lib_occambsd.sh`

`github.com/michaeldexter/occambsd`

Generate a `src.conf`, `KERNCONF`,
build for bhyve, Xen and Jail

Now pulling in a function with `lib_occambsd.sh`!

Software Lesson 18: Position-independent builds

Hold that thought...

```
MAKEOBJDIRPREFIX=/my/object/dir \
```

```
SRCCONF=/my/src.conf \
```

```
make -C /my/source/tree \
```

```
-j48 buildworld
```

```
... DESTDIR=/my/dest/dir installworld ...
```

Software Lesson 19: Some Git for the rest of us

Developers and Ordinary Users, eh?

```
git -C /repo/MAIN clone --mirror \  
https://git.freebsd.org/src.git
```

```
git -C /repo/MAIN remote update
```

Software Lesson 20: Some Git for the rest of us

```
git -C /repo/13.0R clone -b releng/13.0 \  
    file:///repo/MAIN/src.git
```

```
git -C /repo/13.0R/src branch  
    * releng/13.0
```

Software Lesson 21: Some Git for the rest of us

Check out a specific commit/point in time:

```
git -C /repo/13.0R/src checkout \  
Ea31abc261ffc01b6ff5671bffb15cf910a07f4b
```

Just like SVN, `tar -xf ...` Every tutorial says that.
However...

Software Lesson 22: Some Git for the rest of us

```
git -C /repo/13.0R/src log --reverse
--format="%at%x09%H%x09%s" \
releng/13.0 ^origin | nl -v 0 | sed 's/^ *//g' >
src.log
```

0	1611273986	bfd15705156b0436cfe79aea11868dcc0c6e078a	Create the stable/13 branch
1	1611275287	02611ef8ee9f4572d251383560bd6f264f628fe7	Turn on MALLOC_PRODUCTION
2	1611310654	d5bd29ed505f535b52673604951d018a43d72579	Revert "[mips] revert r366664 -
flip mips back from -O2 to -O"			
...			
404	1617927270	ea31abc261ffc01b6ff5671bffb15cf910a07f4b	13.0: update to RELEASE
405	1618670495	a8a91efa74e09f35ecde161f6bb18f10e28efddf	mpt(4): Remove incorrect S/G

Software Lesson 23: Meta Mode

```
kldload filemon
```

```
WITH_META_MODE=YES make buildworld
```

That's it.

First build is “full”, subsequent are deltas.

Software Lesson 24: up.bsd.lv and headbanger.sh

"Tracking branches should not be a rite of passage"

up.bsd.lv

Great work Conor!

`github.com/michaeldexter/headbanger`

tl:dr: `freebsd-update` is not named `freebsd-upgrade`...

"That's how we've always done it."

Software Lesson 25: BUT... Moar Reflection

pkg'd base please!

freebsd-update (f-u) appeared prior to...

Meta Mode builds

Reproducible Builds

OpenZFS

Git, EPYC CPUs

The Ransomware Epidemic...

Software Lesson 26: What if...

What if builds were cheap? Like, really cheap?

f-u: DVD1.iso -> init.sh ... stuff ... OMG no.

“Nope it from orbit” – Allan Jude

Meta Mode, Reproducible Builds, and OpenZFS
make deltas *really* cheap.

Software Lesson 27: What if...

- Meta Mode + Reproducible Builds + OpenZFS make deltas *really* cheap
- EPYC + OccamBSD make builds cheap and *fast*
- Git makes branch navigation fast
- OpenZFS makes immutability really cheap and dynamic
- Ransomware... *really* makes updating and immutability a priority

Software Lesson 28: So...

Snapshot every commit on a branch?

ZFS don' care.

Snapshot every meta mode build on a branch?

ZFS don' care.

Diff and clone differences to clone?

ZFS don' care.

Software Lesson 29: So...

Read-only kernel, binary, and library datasets?

ZFS don' care.

Slam the binary differences into Git?

Git don' care.

Slam the binary differences into SVN?

`svn-lite` don' care.

Software Lesson 30: So...

You have 80% of what you need for an atomic and immutable FreeBSD upgrade mechanism that can traverse all branches, commit by commit, on ZFS and UFS-based systems

`github.com/michaeldexter/petribsd`

Seriously, Thank You EuroBSDcon!

Merci! Paldies! Danke! Gracias! Tack!|Takk!|Tak! Grazie! Ευχαριστώ! Ačiū! Dziękuję! Eskerrik asko!

Michael Dexter

@michaeldexter
editor@callfortesting.org